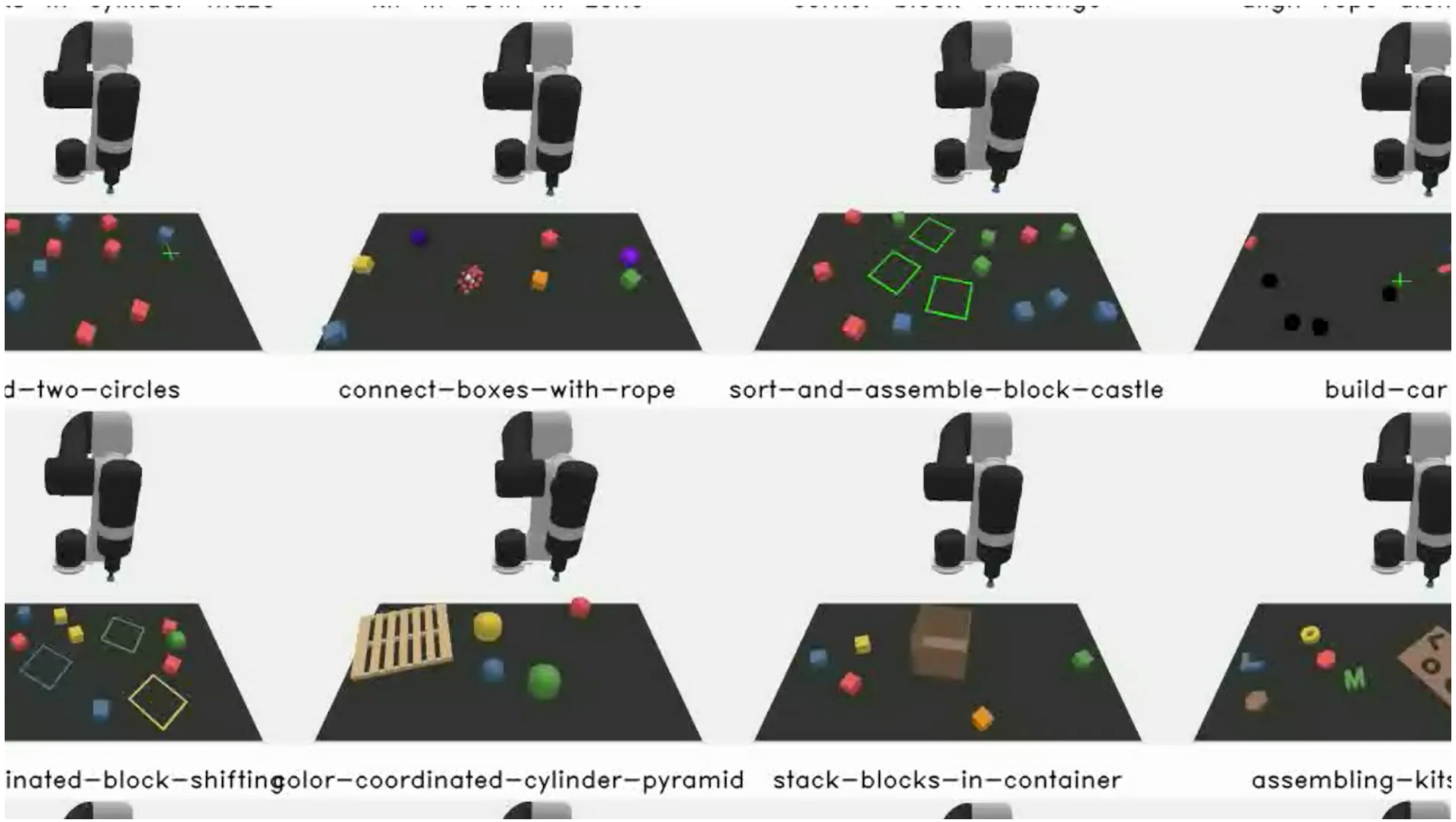
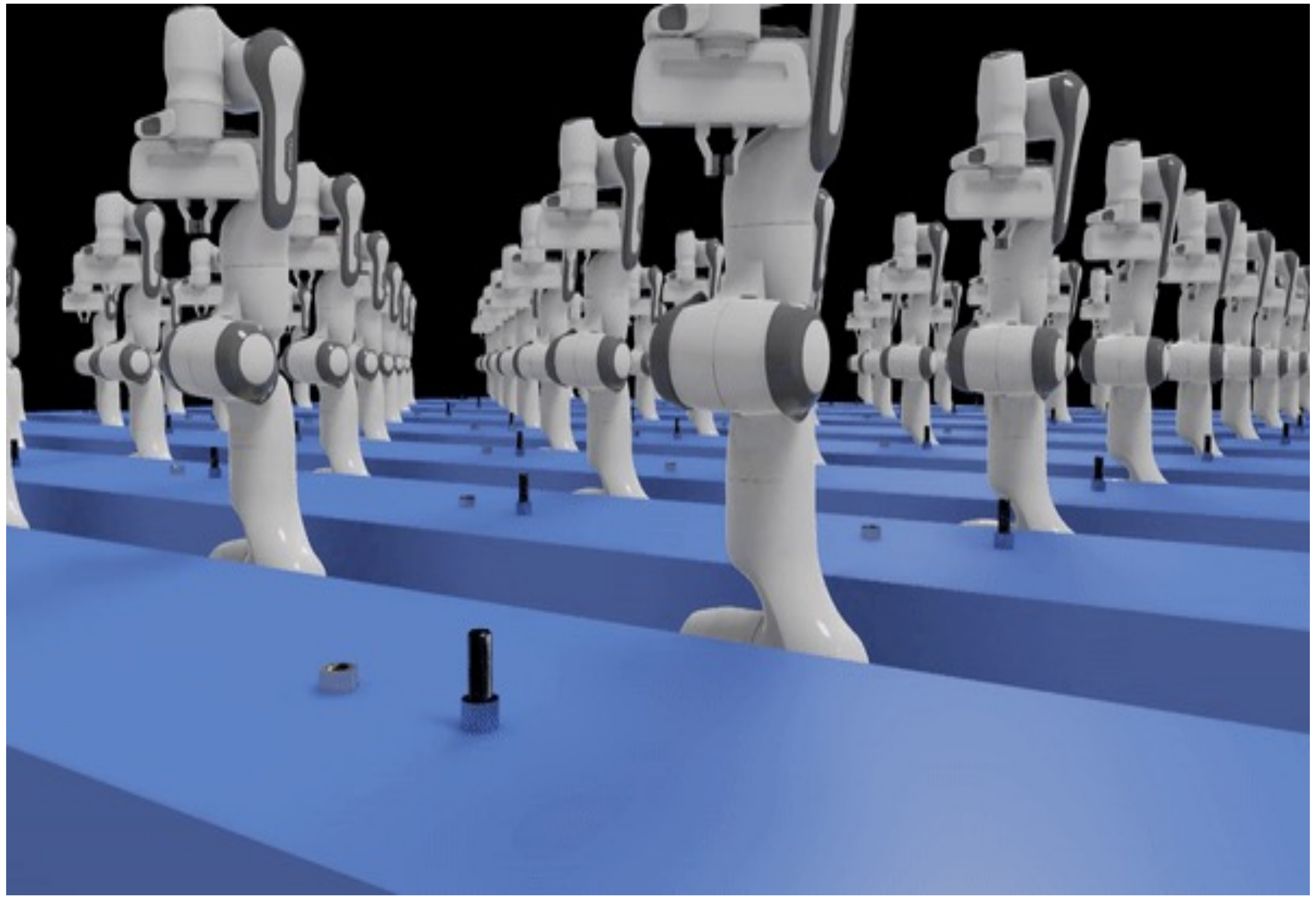
# **GenSim**: Generating Robotic Simulation Tasks via Large Language Models

**Lirui Wang**, Yiyang Ling*, Zhecheng Yuan*, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, Xiaolong Wang

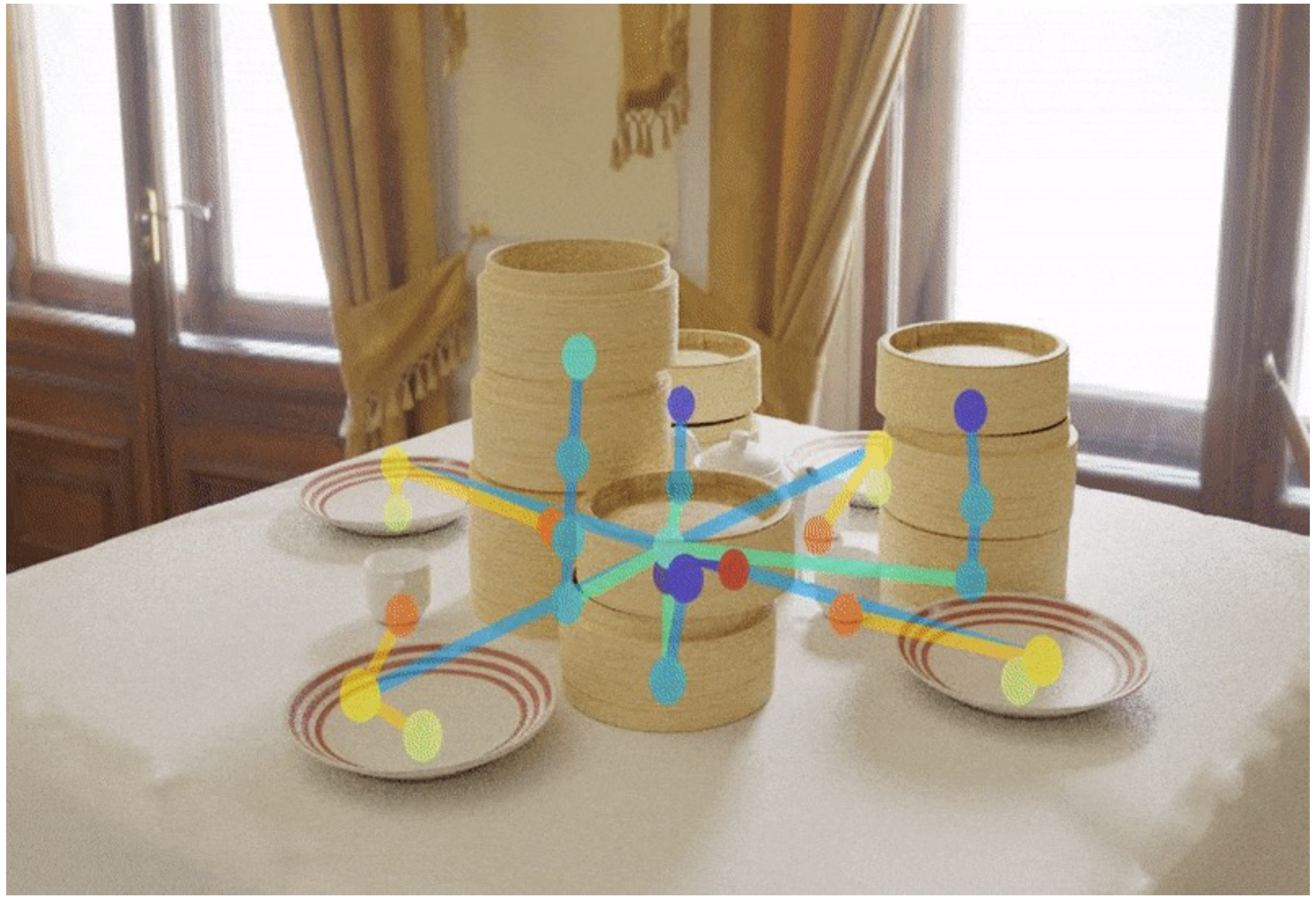
**ICLR 2024**


d−two−circles
connect−boxes−with−rope
sort−and−assemble−block−castle
build−car
inated−block−shifting
color−coordinated−cylinder−pyramid
stack−blocks−in−container
assembling−kits

# Current Methods to Scale Up Data in Simulation



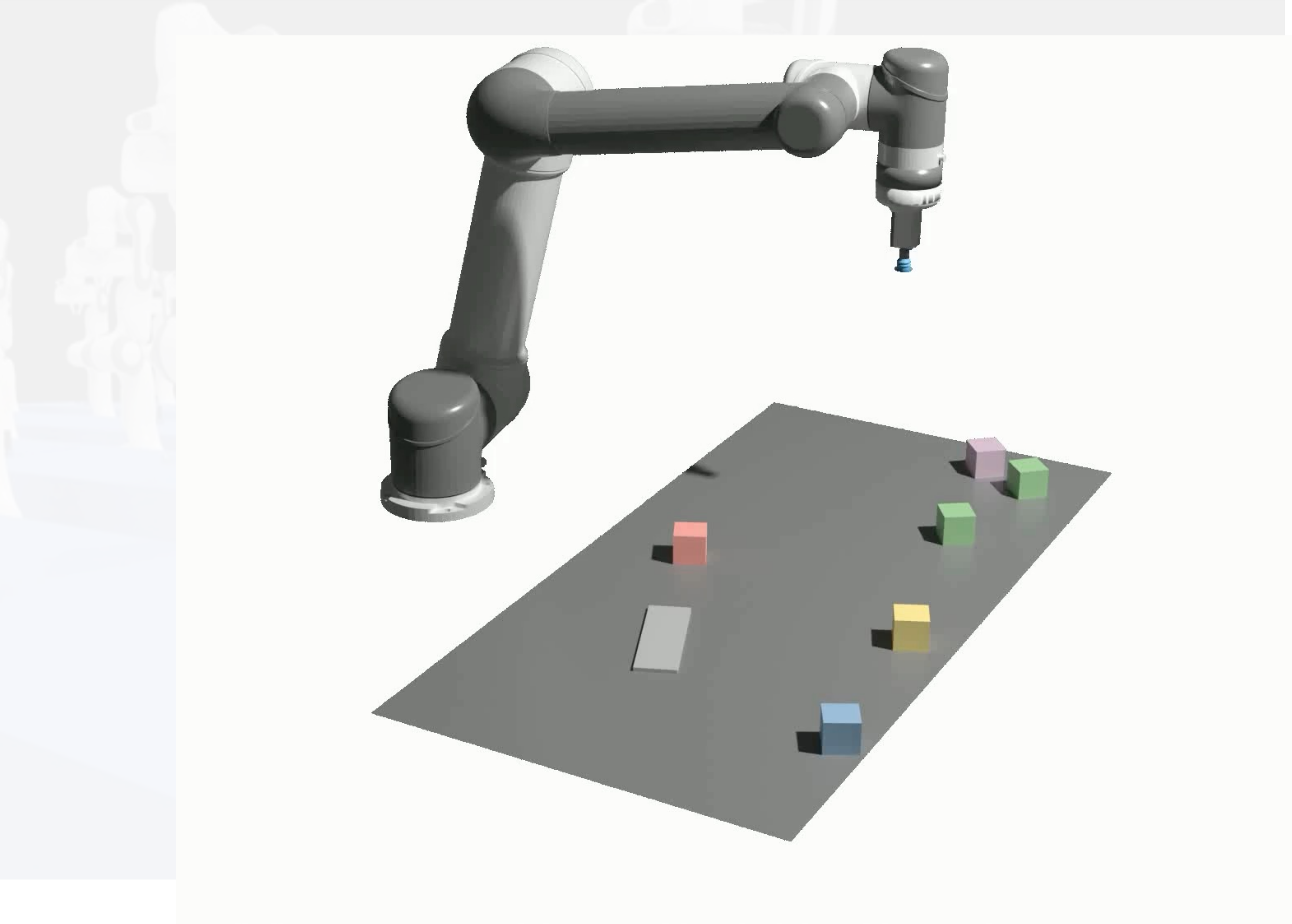
Massive Rollouts



Randomized Scenes

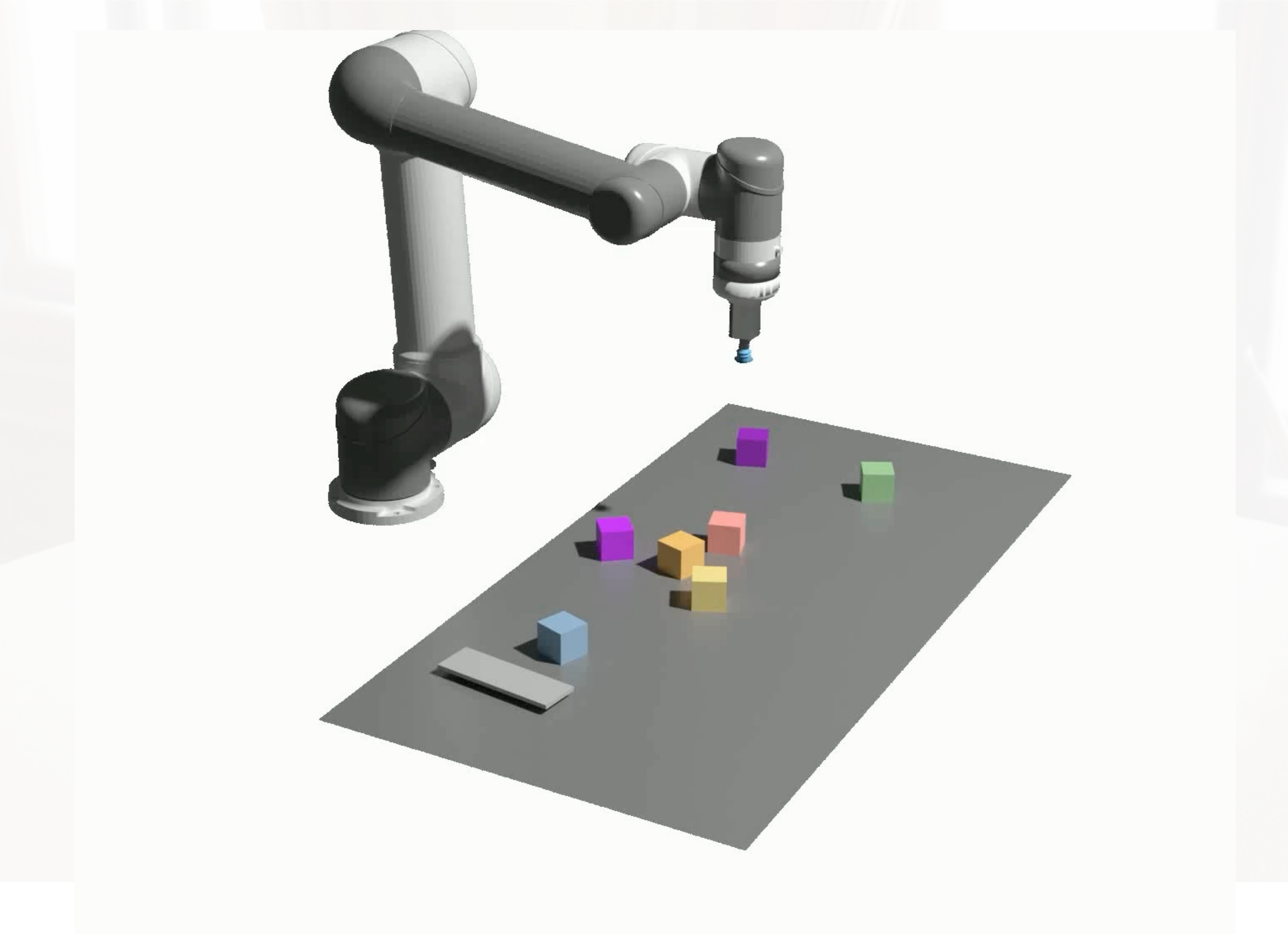
*Current Methods to Scale Up Data in Simulation*

# Lack of Task Diversity



Massive Parallel Rollouts

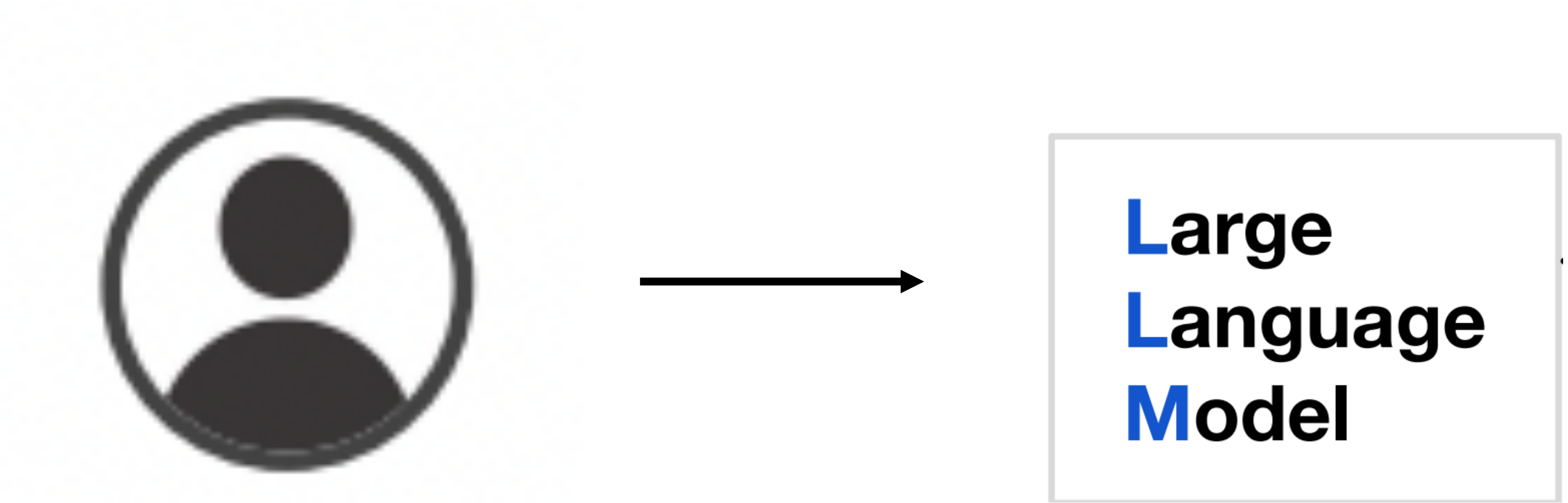
Stack-block-pyramid



Randomized Domains

Stack-rainbow-tower

# GenSim Framework



Large
Language
Model

Can you generate the task “build-car”?

Task Creator

# GenSim Framework

Can you generate the task "build-car"?
Large Language Model
Task Creator
Program Synthesis

```
class BuildCar(Task):
    """Construct a simple car structure using blocks and cylinders."""

     ...
    def reset(self, env):
   # Add wheels.
   wheel_size = (0.02, 0.02, 0.02)  # x, y, z dimensions for the asset size
   wheel_urdf = 'cylinder/cylinder-template.urdf'

     ...
    for idx in range(4):
        wheel_pose = self.get_random_pose(env, wheel_size)
        wheel_id = env.add_object(wheel_urdf, wheel_pose, color=utils.COLORS['black'])
        wheels.append(wheel_id)
     ...

    self.add_goal(
      objs=wheels,
      matches=np.ones((4, 4)),
      ...
      language_goal="For the wheels, place a black cylinder on each side of the base blocks.")
```

# GenSim Framework

Can you generate the task “build-car”?

Large Language Model

Task Creator

Program Synthesis

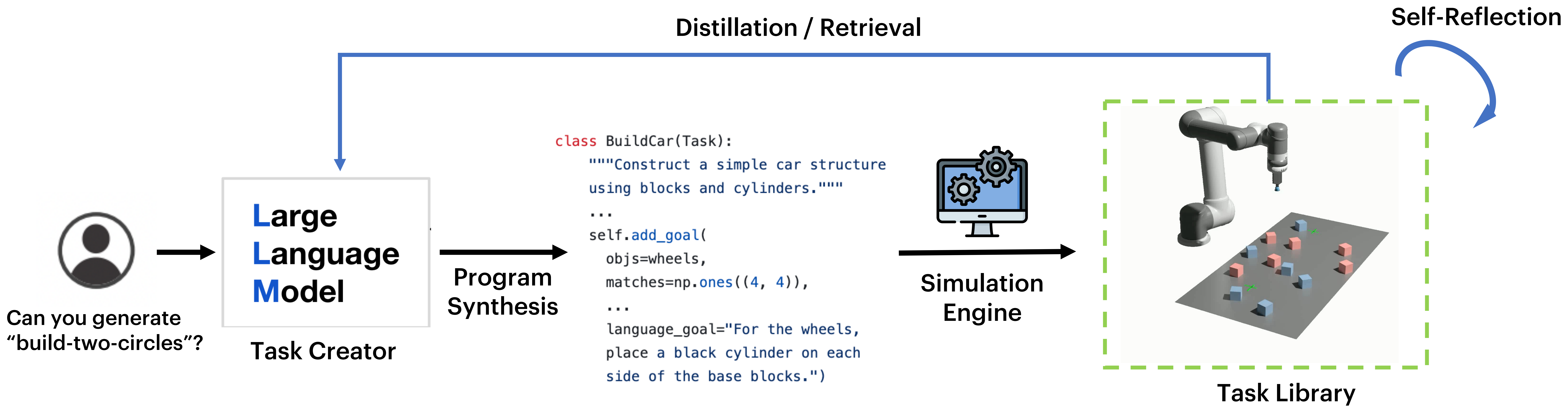
```
class BuildCar(Task):
    """Construct a simple car structure
    using blocks and cylinders."""
    ...
    self.add_goal(
      objs=wheels,
      matches=np.ones((4, 4)),
      ...
      language_goal="For the wheels,
      place a black cylinder on each
      side of the base blocks.")
```

Simulation Engine

Does the code implement “build-car”

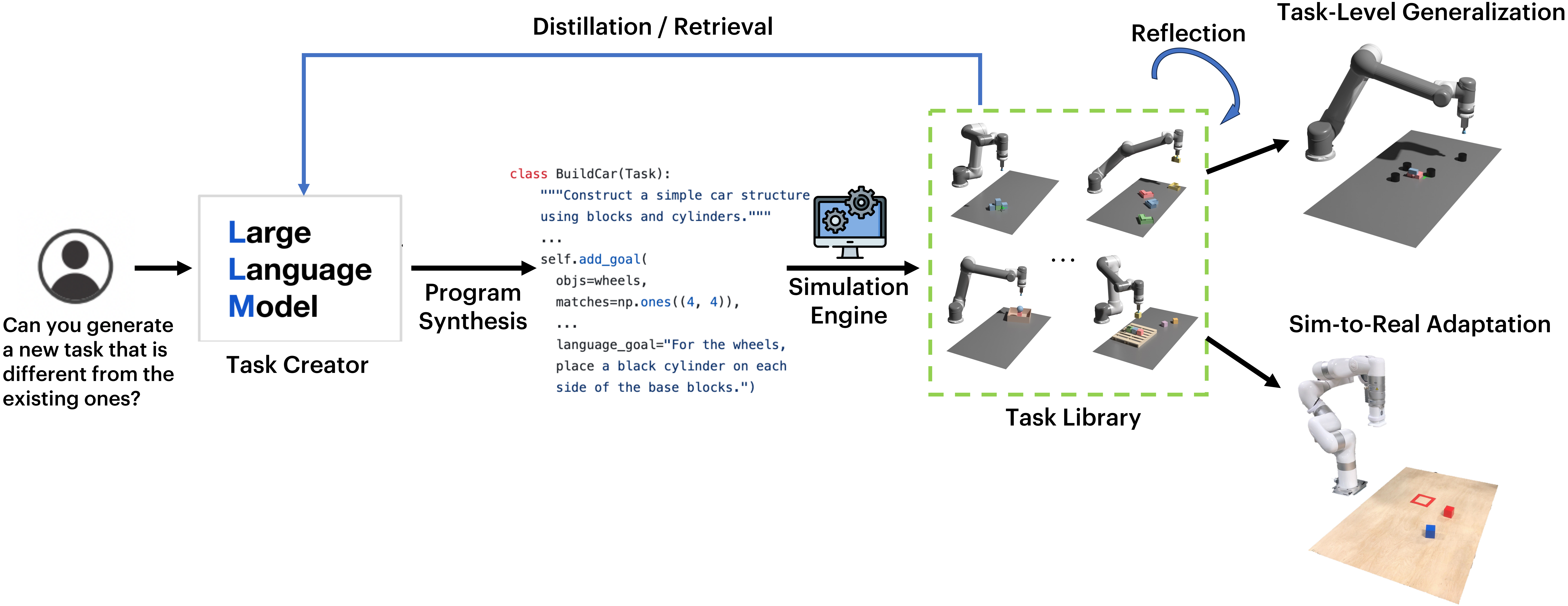
Self-Reflection

# GenSim Framework



```
class BuildCar(Task):
    """Construct a simple car structure
    using blocks and cylinders."""
    ...
    self.add_goal(
      objs=wheels,
      matches=np.ones((4, 4)),
      ...
      language_goal="For the wheels,
      place a black cylinder on each
      side of the base blocks.")
```

# GenSim Framework



Distillation / Retrieval

Can you generate a new task that is different from the existing ones?

Large Language Model

Task Creator

Program Synthesis

```
class BuildCar(Task):
    """Construct a simple car structure
    using blocks and cylinders."""
    ...
    self.add_goal(
      objs=wheels,
      matches=np.ones((4, 4)),
      ...
      language_goal="For the wheels,
      place a black cylinder on each
      side of the base blocks.")
```

Simulation Engine

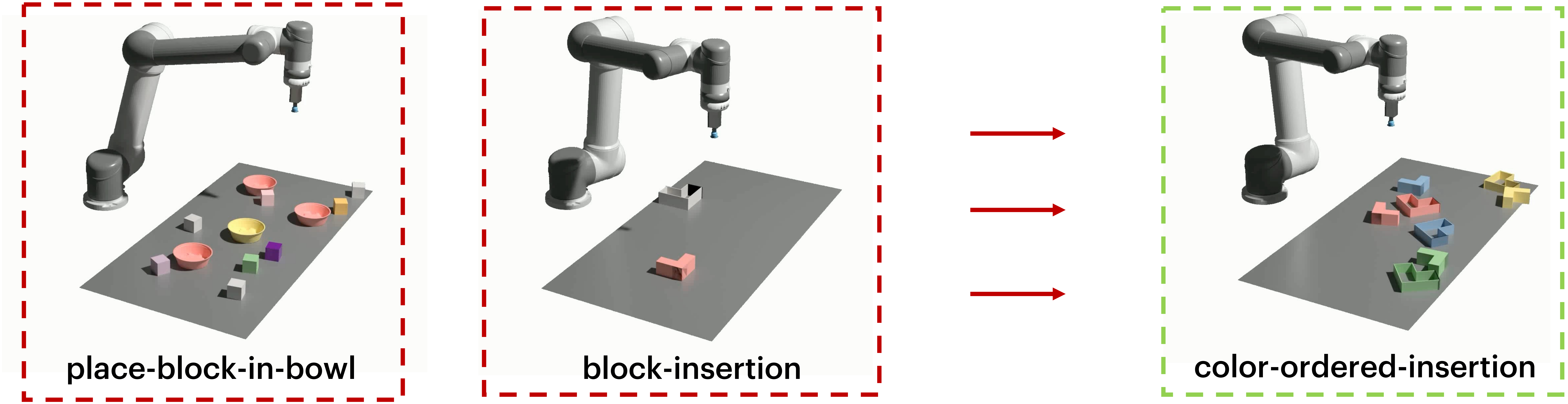
Task Library

Reflection

Task-Level Generalization

Sim-to-Real Adaptation

# Generated Tasks



place-block-in-bowl

block-insertion

color-ordered-insertion

# Generated Tasks

composition

extrapolation

stack-block-pyramid

palletizing-boxes

pyramid-on-pallet

rainbow-stack

build-bridge

construct-corner-blocks

build-car

place-block-in-bowl

block-insertion

color-ordered-insertion

ball-on-box-in-container

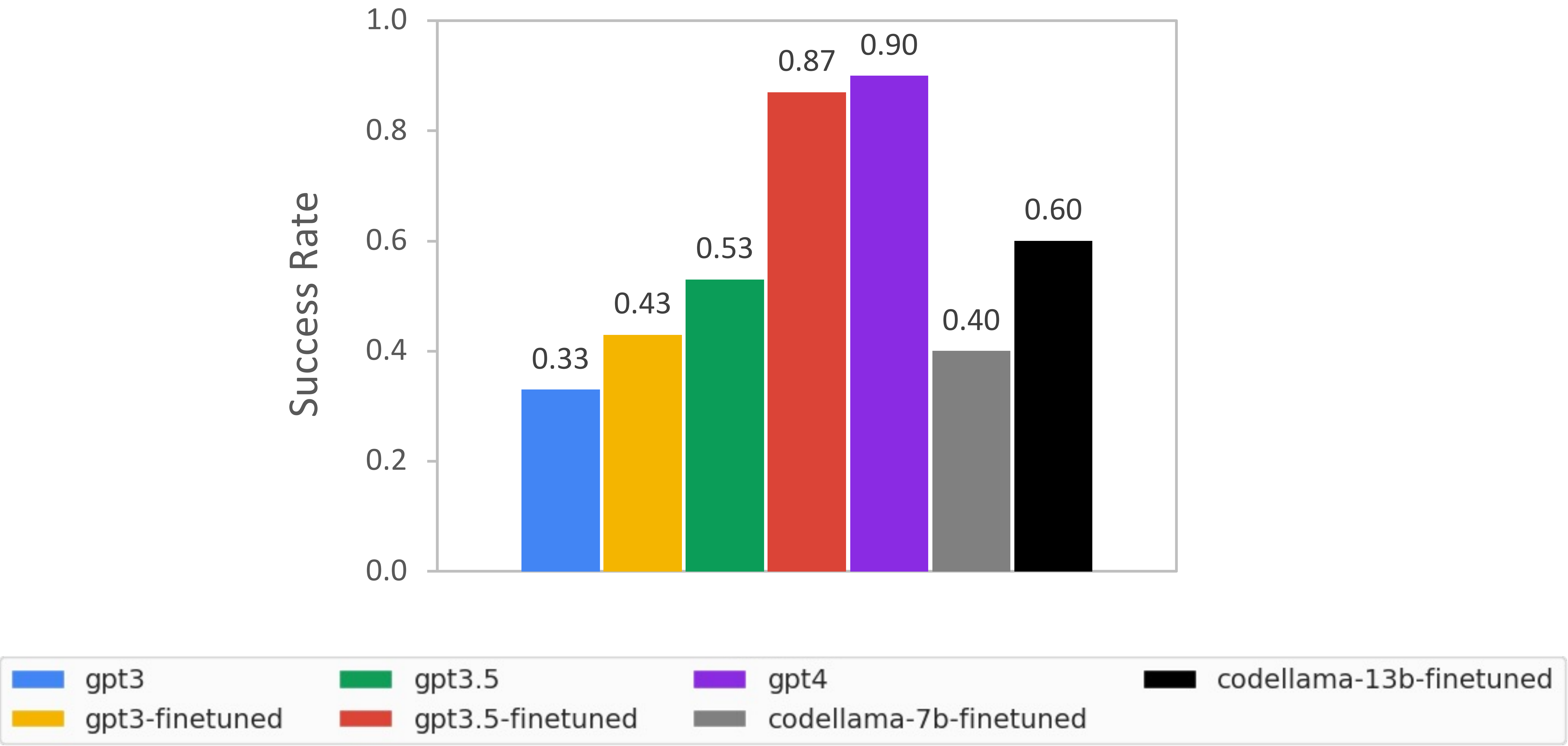
pair-blocks-along-line

build-two-circles

four-corner-pyramid

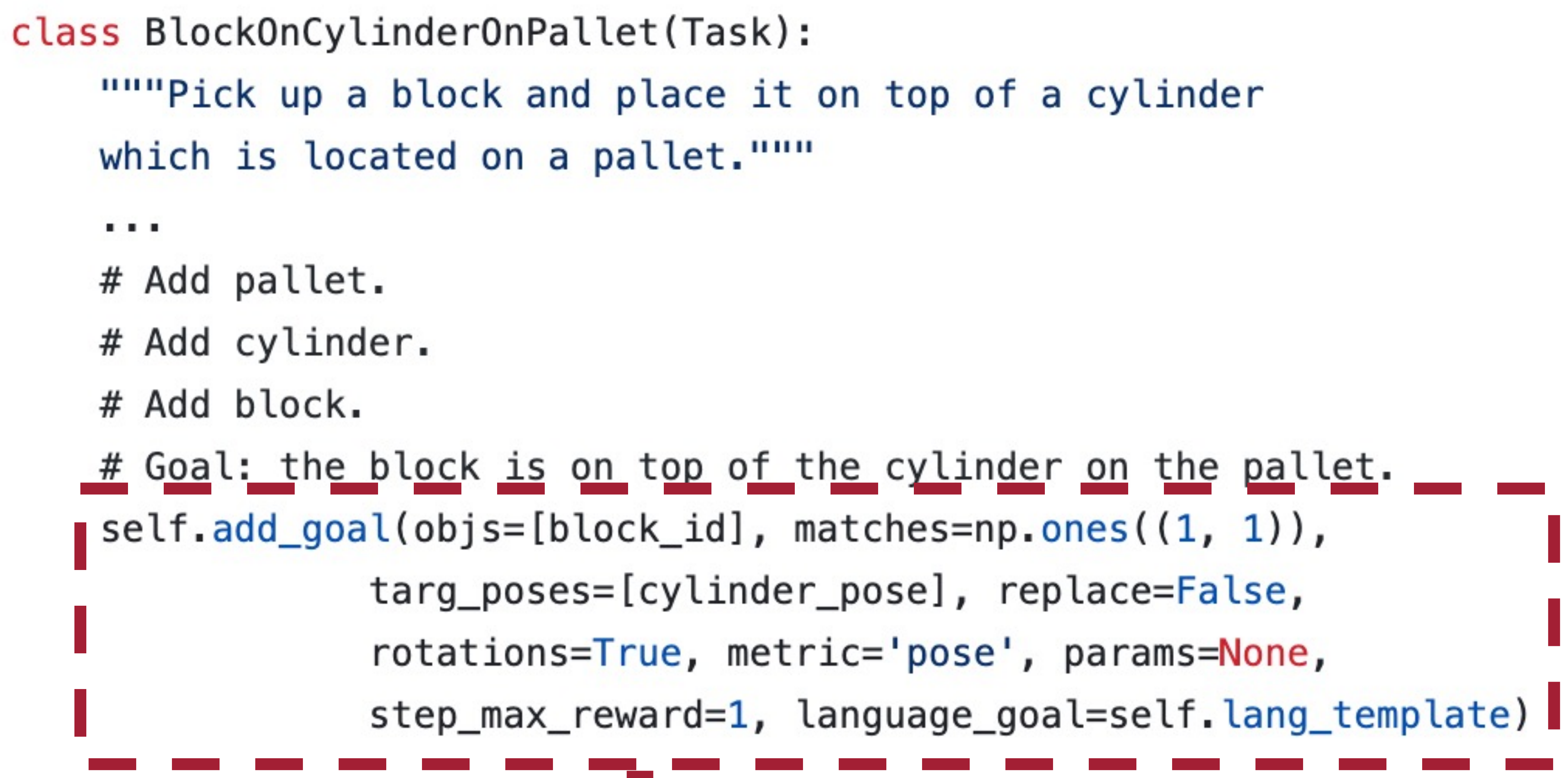
# Experiments

# LLM Task Generation Evaluation



# Qualitative Comparison

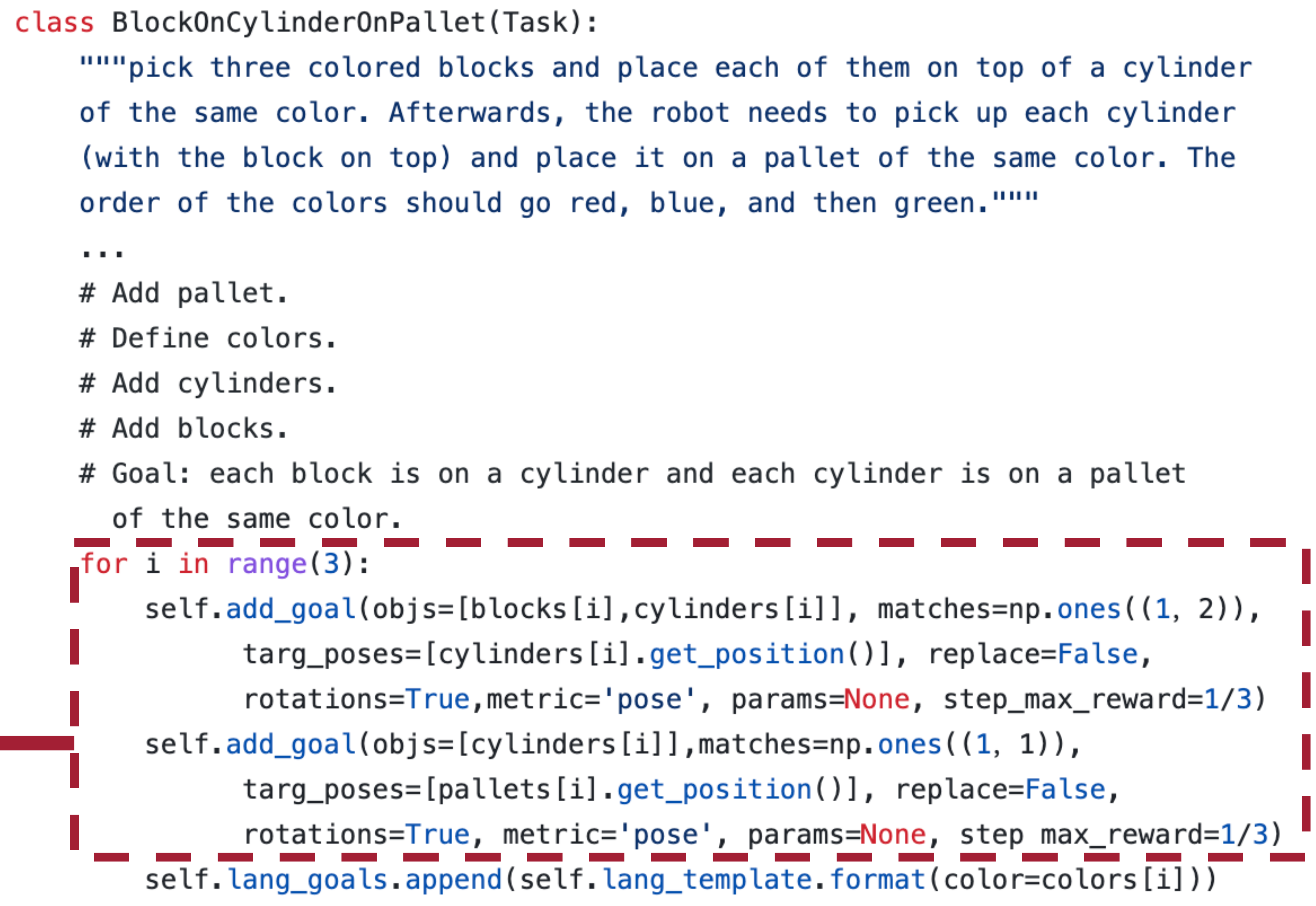
## Code-Llama (Finetuned)



```
class BlockOnCylinderOnPallet(Task):
    """Pick up a block and place it on top of a cylinder
    which is located on a pallet."""
    ...
    # Add pallet.
    # Add cylinder.
    # Add block.
    # Goal: the block is on top of the cylinder on the pallet.
    self.add_goal(objs=[block_id], matches=np.ones((1, 1)),
                 targ_poses=[cylinder_pose], replace=False,
                 rotations=True, metric='pose', params=None,
                 step_max_reward=1, language_goal=self.lang_template)
```

No pallet

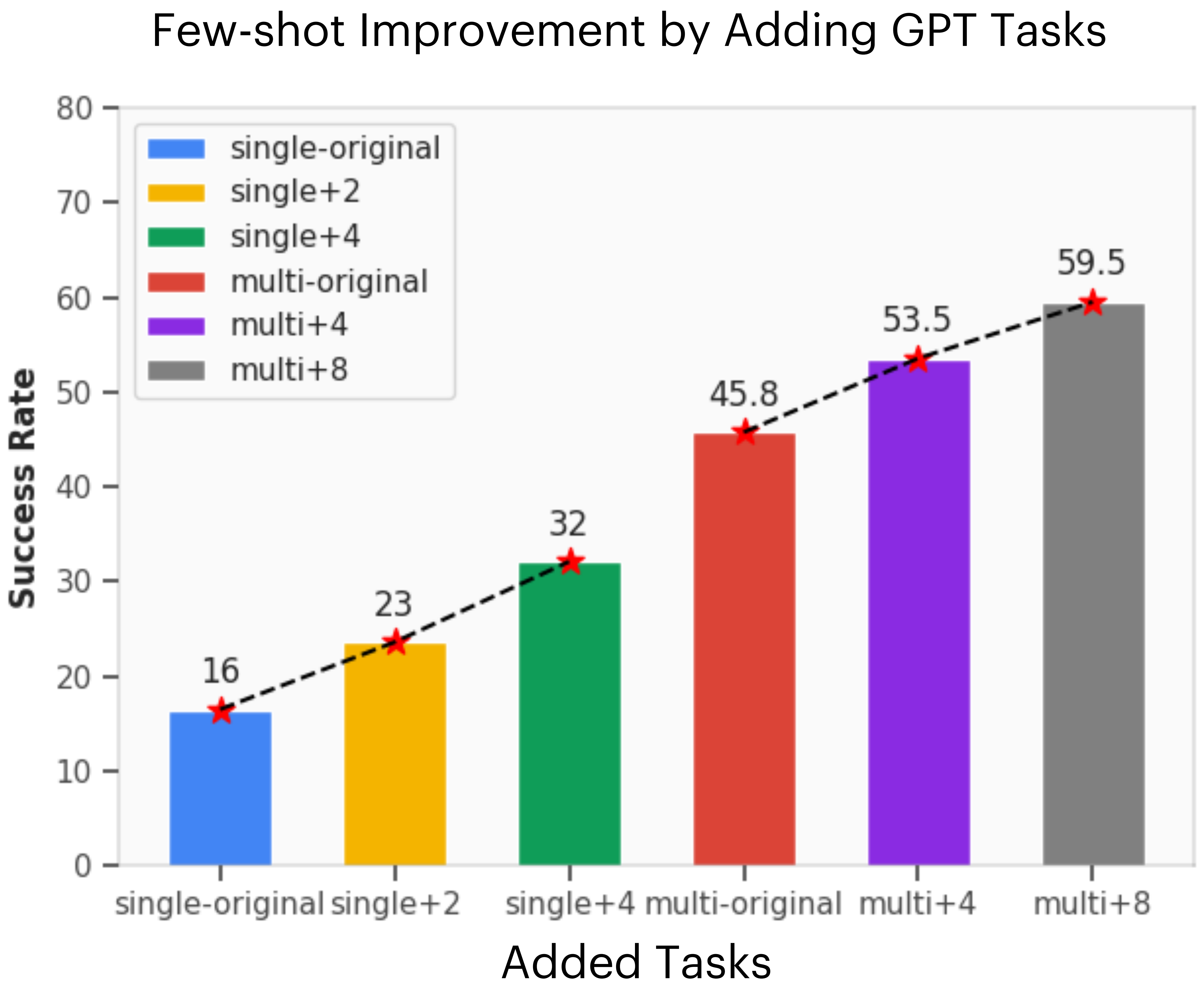
## GPT-4

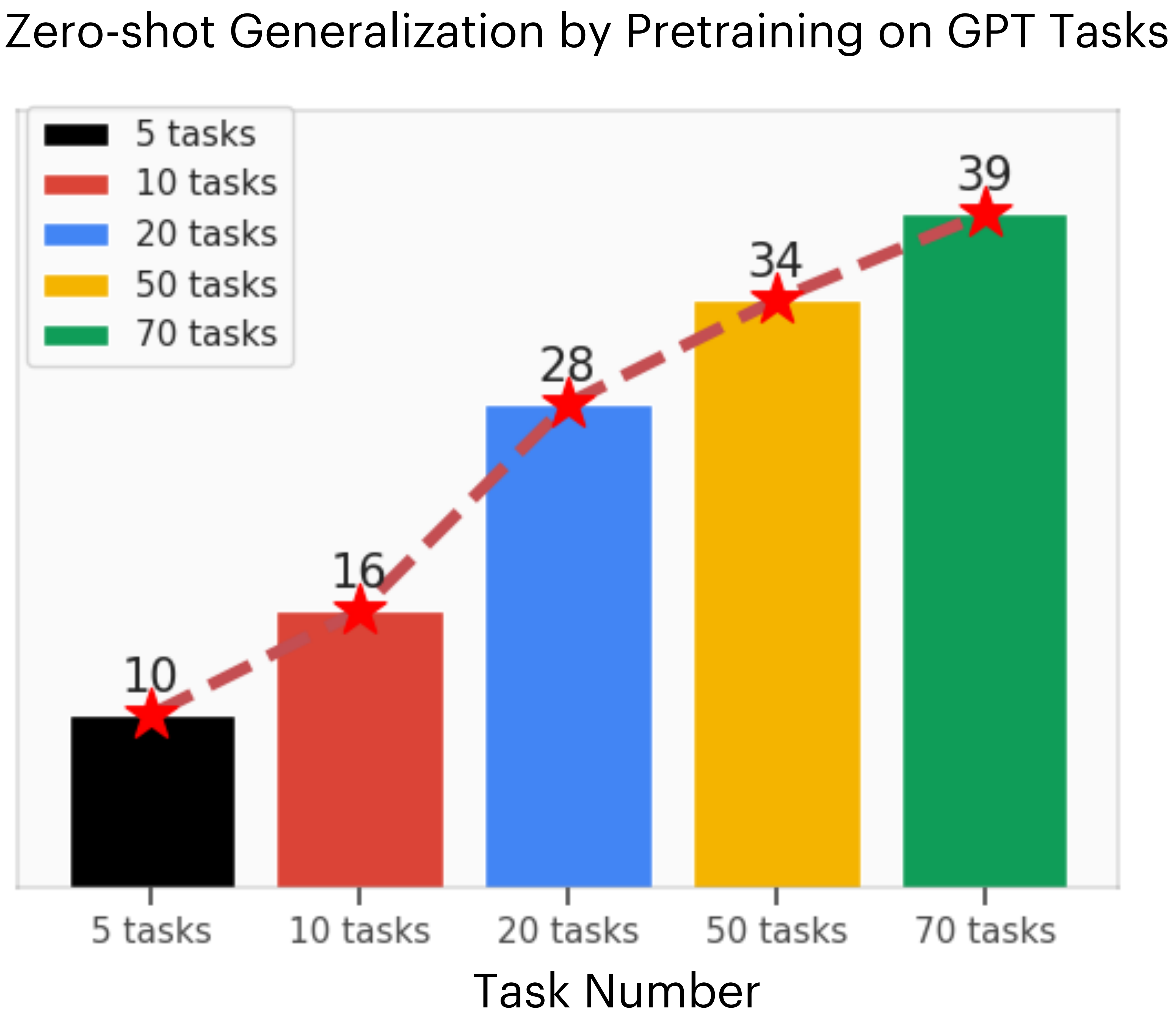


```
class BlockOnCylinderOnPallet(Task):
    """pick three colored blocks and place each of them on top of a cylinder
    of the same color. Afterwards, the robot needs to pick up each cylinder
    (with the block on top) and place it on a pallet of the same color. The
    order of the colors should go red, blue, and then green."""
    ...
    # Add pallet.
    # Define colors.
    # Add cylinders.
    # Add blocks.
    # Goal: each block is on a cylinder and each cylinder is on a pallet
      of the same color.
    for i in range(3):
        self.add_goal(objs=[blocks[i],cylinders[i]], matches=np.ones((1, 2)),
              targ_poses=[cylinders[i].get_position()], replace=False,
              rotations=True,metric='pose', params=None, step_max_reward=1/3)
        self.add_goal(objs=[cylinders[i]],matches=np.ones((1, 1)),
              targ_poses=[pallets[i].get_position()], replace=False,
              rotations=True, metric='pose', params=None, step_max_reward=1/3)
        self.lang_goals.append(self.lang_template.format(color=colors[i]))
```

Wrong order

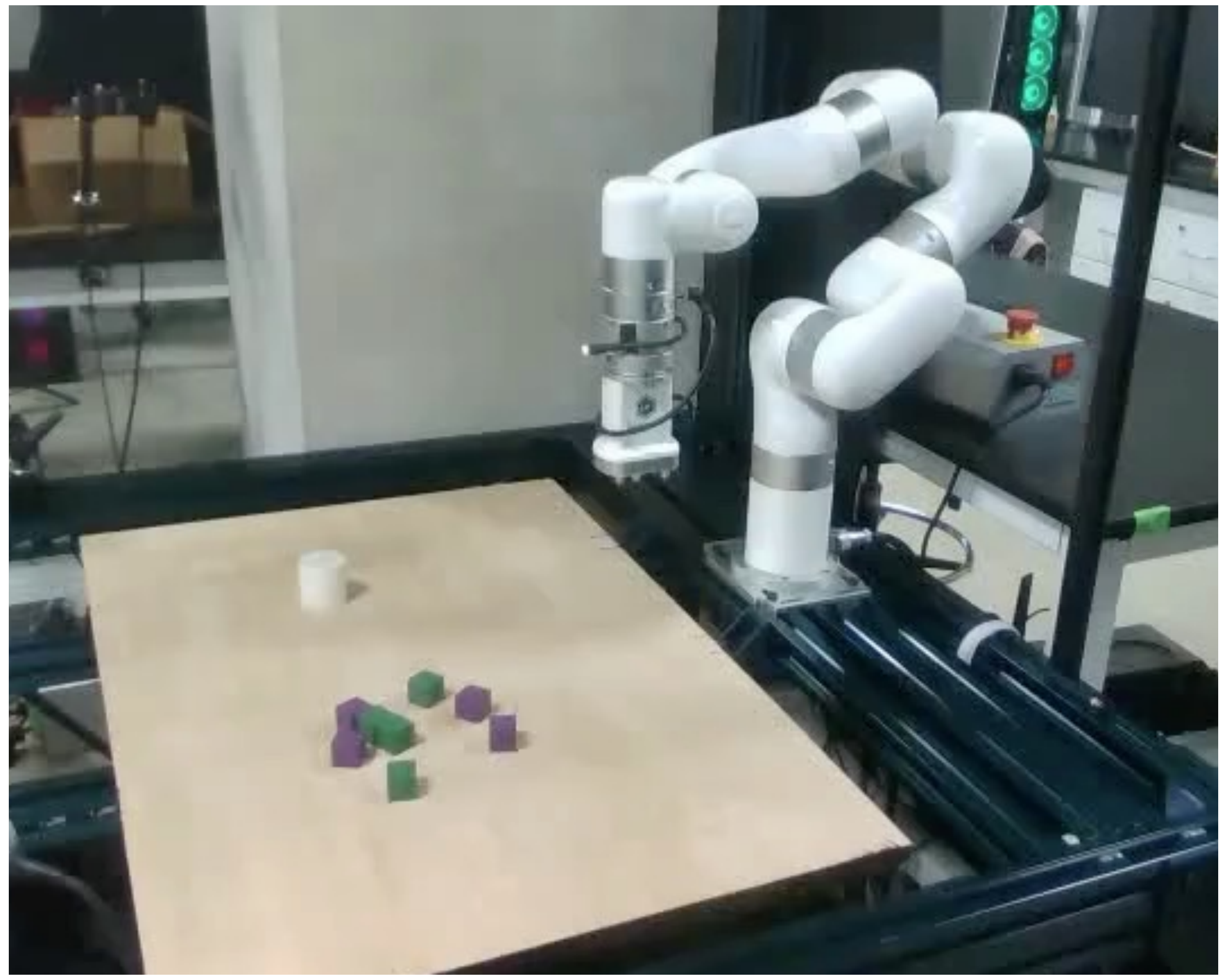
# Multitask Simulation Policy Training

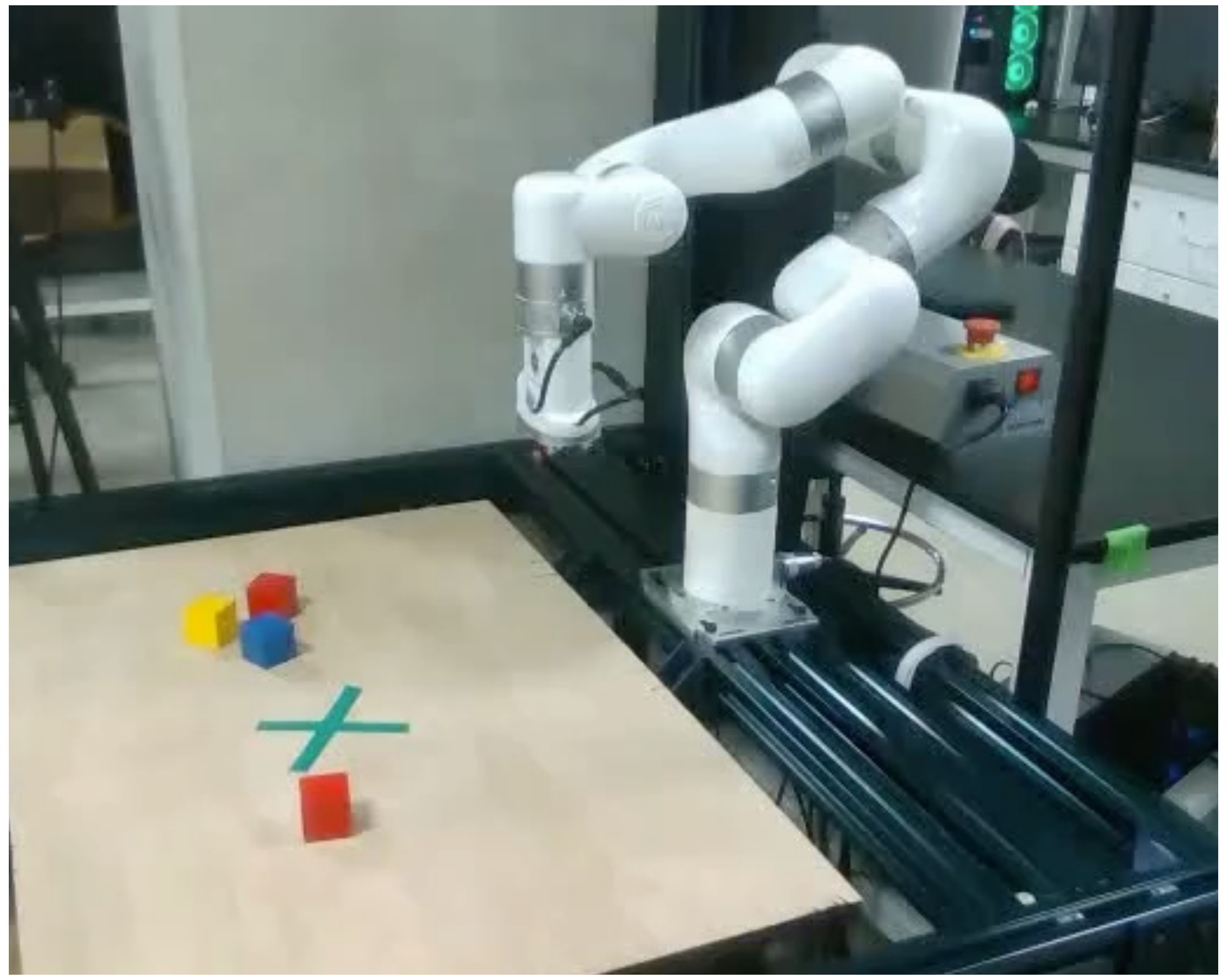




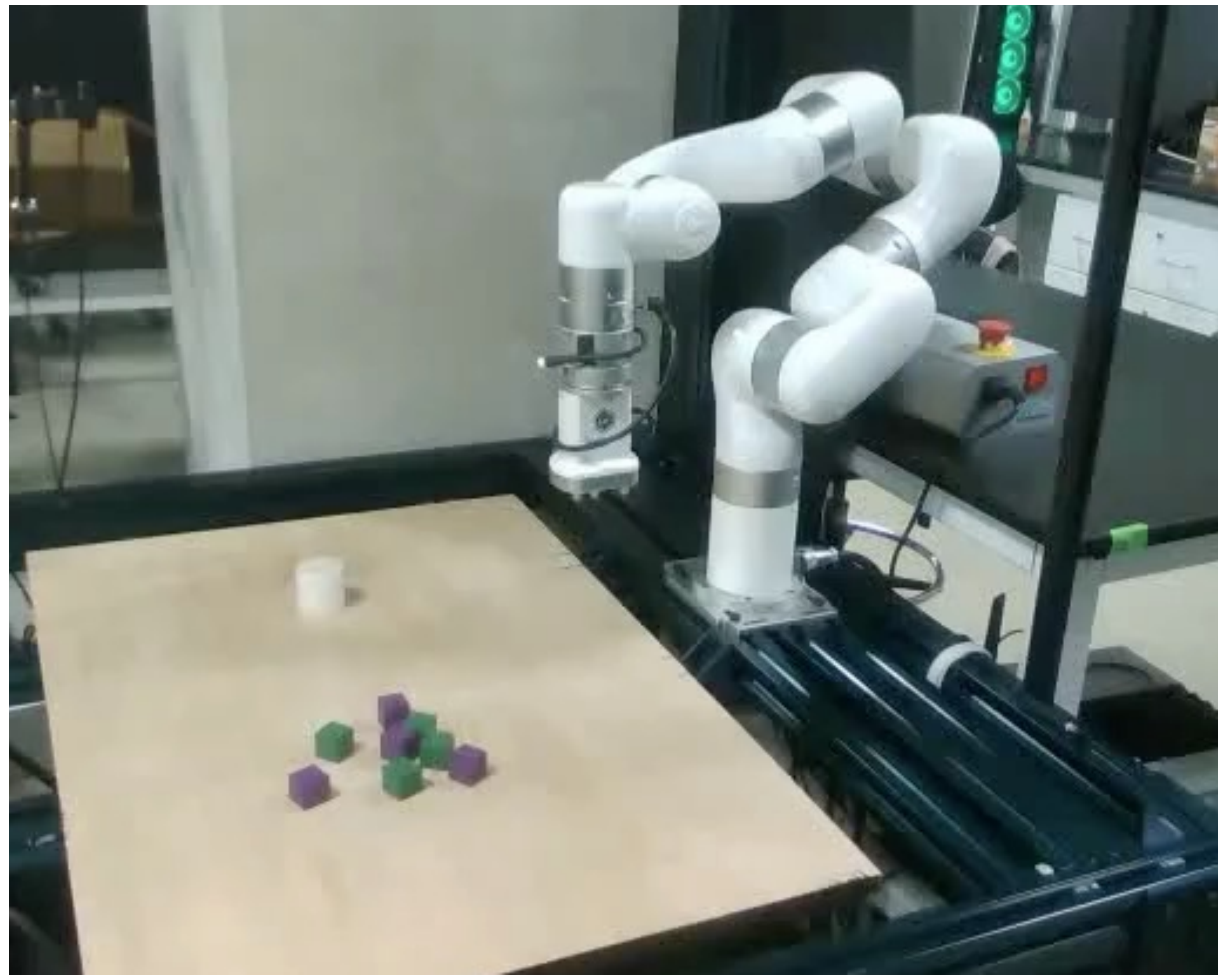
# Real-World Policy Rollouts (4x)

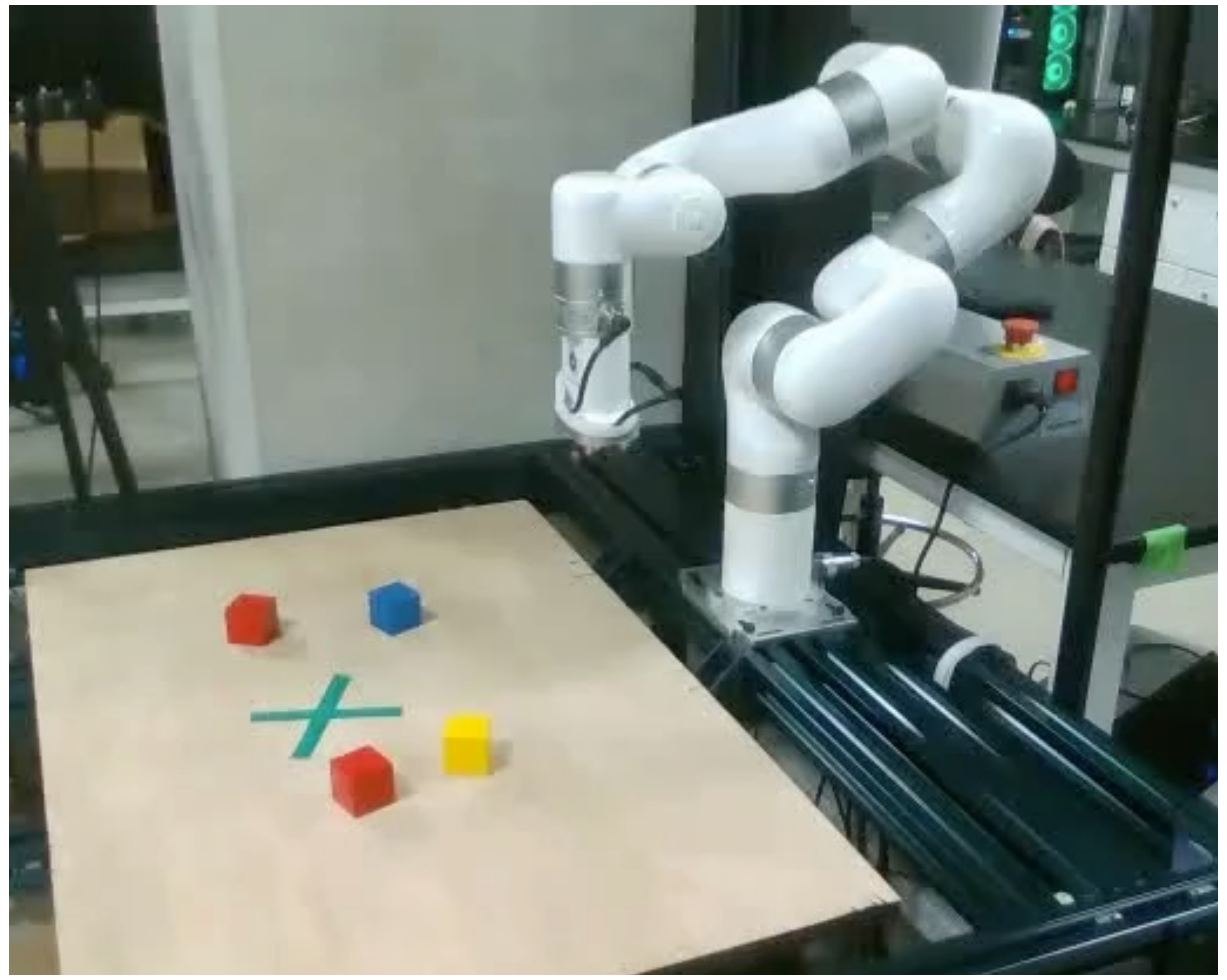
CLIPort





Ours





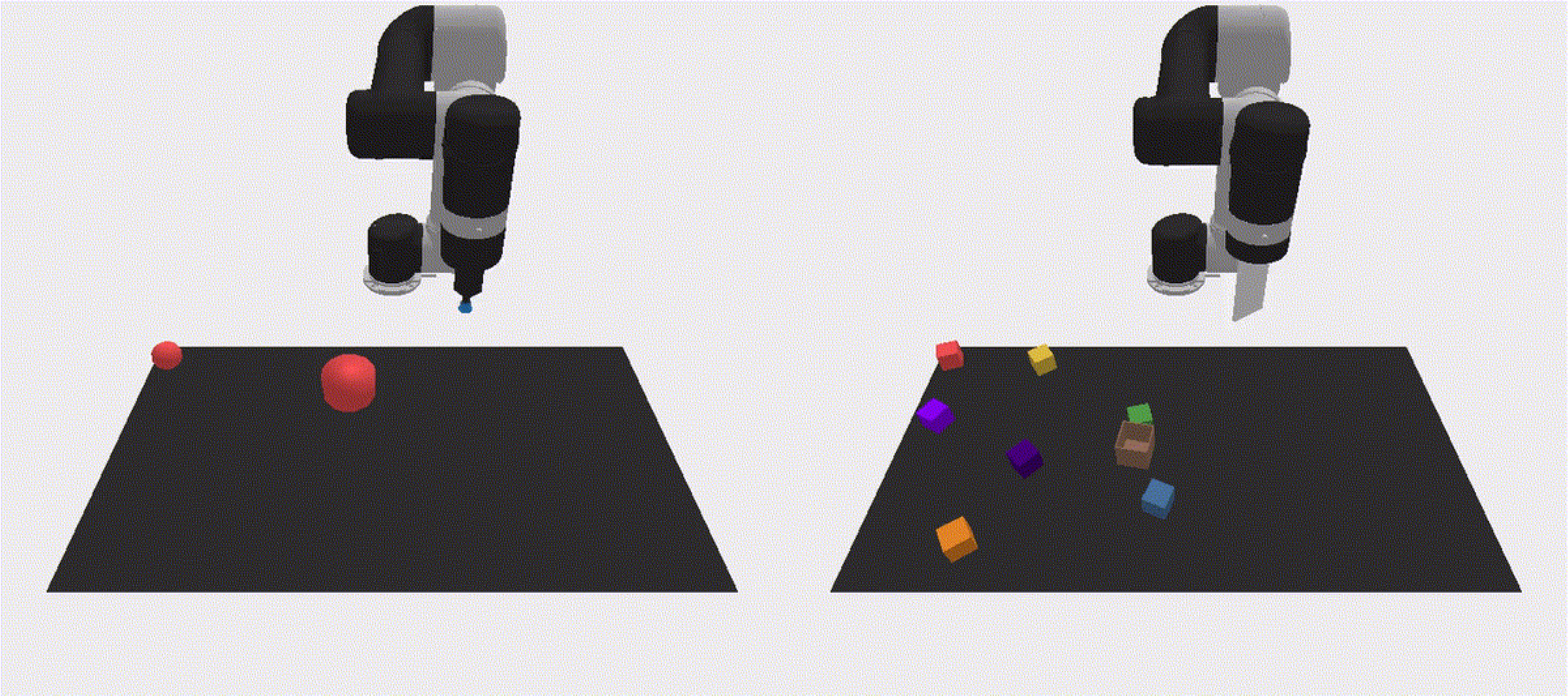
Success Rates: 43.3% 68.8%

## *Limitation*

- Imperfect physical grounding
- Limited task complexity

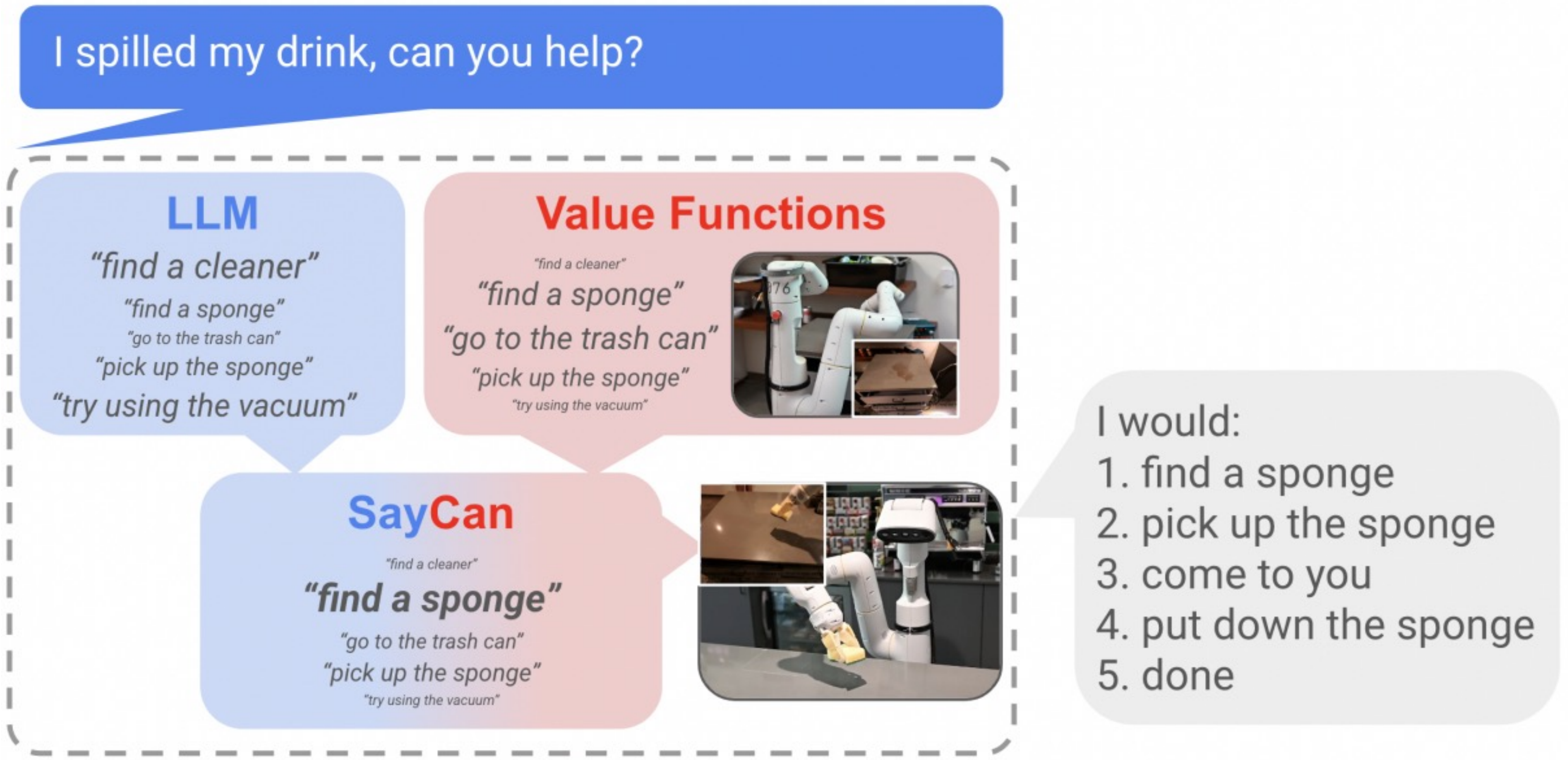
## *Future Work*

- Improve VLM/LLM for Robotics
- Dexterous tasks and asset generation



# Related Works

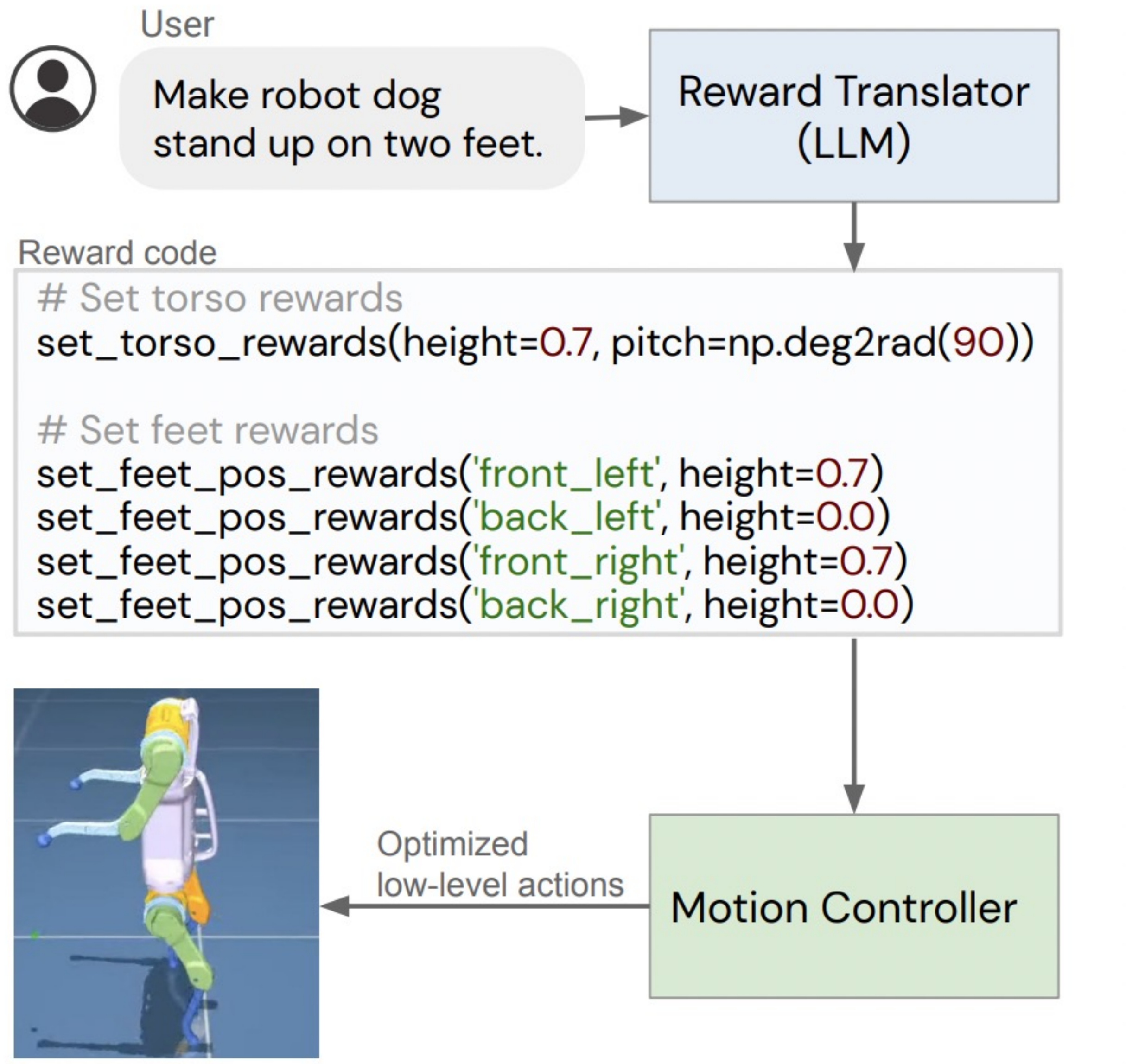
## Say-Can (2022)



I spilled my drink, can you help?

I would:
1. find a sponge
2. pick up the sponge
3. come to you
4. put down the sponge
5. done

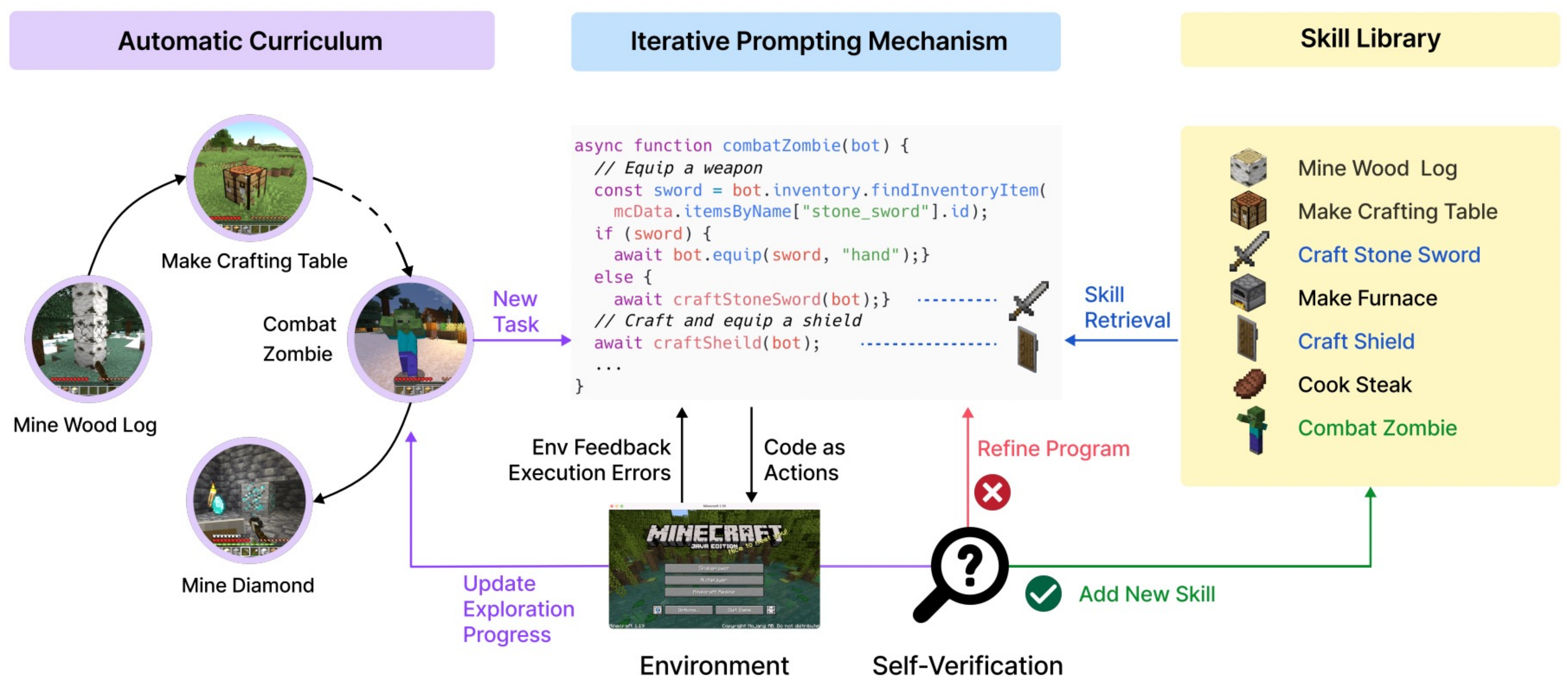
## Language to Reward (2023)



User
Make robot dog stand up on two feet.
Reward Translator (LLM)
Reward code
Optimized low-level actions
Motion Controller

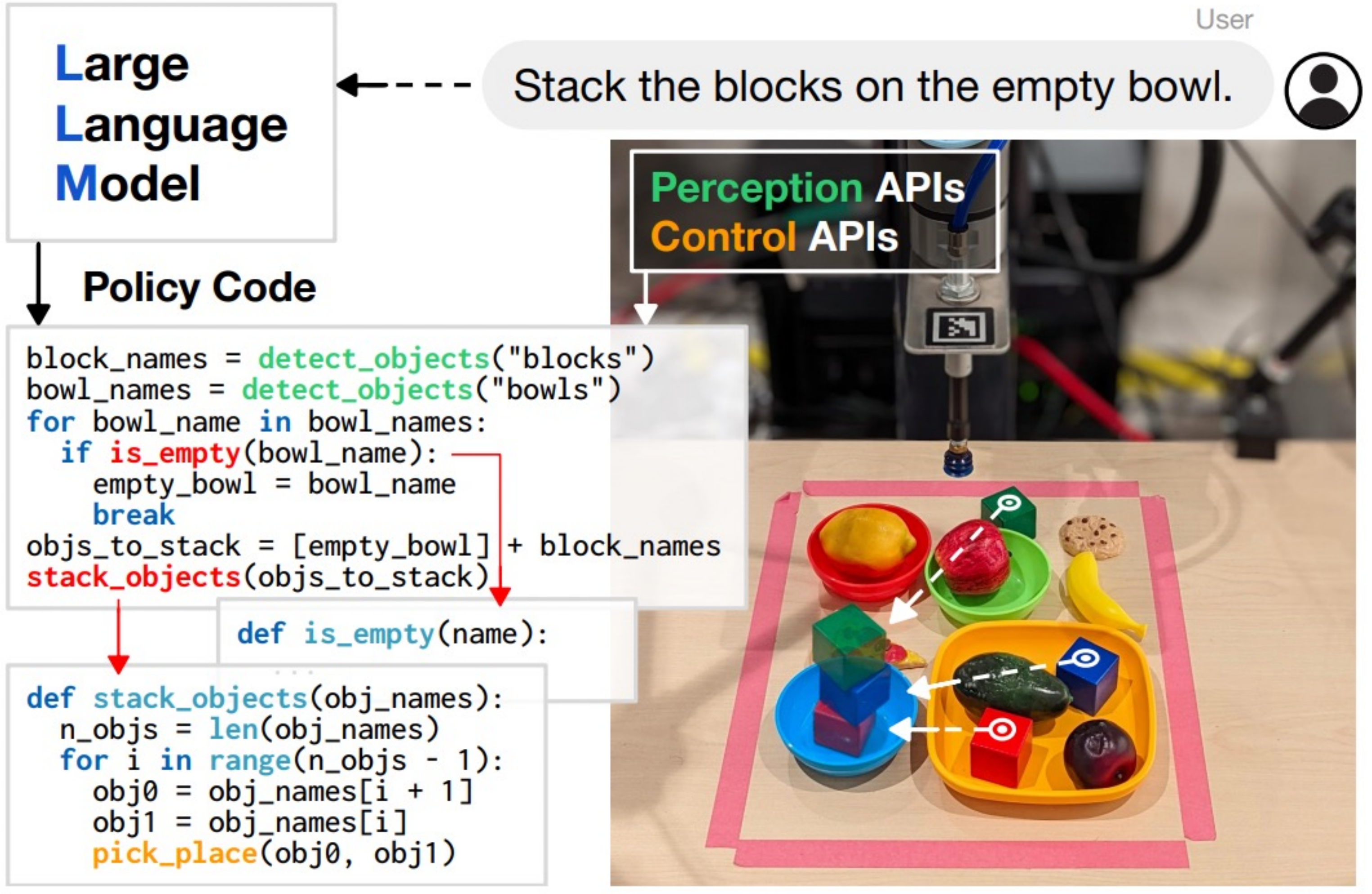
```
# Set torso rewards
set_torso_rewards(height=0.7, pitch=np.deg2rad(90))

# Set feet rewards
set_feet_pos_rewards('front_left', height=0.7)
set_feet_pos_rewards('back_left', height=0.0)
set_feet_pos_rewards('front_right', height=0.7)
set_feet_pos_rewards('back_right', height=0.0)
```

## Voyager (2023)



```
async function combatZombie(bot) {
  // Equip a weapon
  const sword = bot.inventory.findInventoryItem(
    mcData.itemsByName["stone_sword"].id);
  if (sword) {
    await bot.equip(sword, "hand");}
  else {
    await craftStoneSword(bot);}
  // Craft and equip a shield
  await craftSheild(bot);
  ...
}
```

## Code-as-Policies (2022)

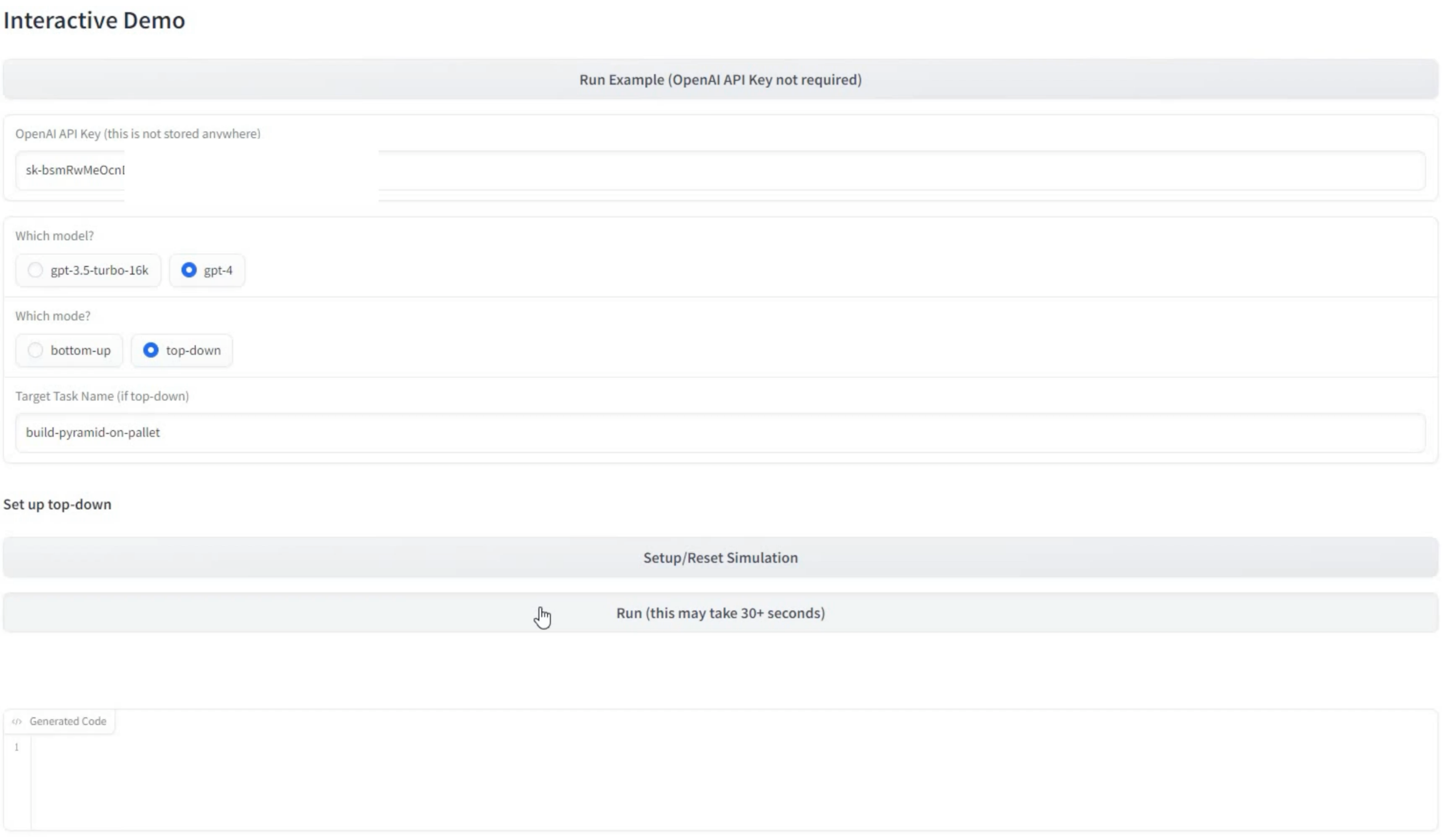


```
block_names = detect_objects("blocks")
bowl_names = detect_objects("bowls")
for bowl_name in bowl_names:
  if is_empty(bowl_name):
    empty_bowl = bowl_name
    break
objs_to_stack = [empty_bowl] + block_names
stack_objects(objs_to_stack)
```

```
def is_empty(name):
  ...
```

```
def stack_objects(obj_names):
  n_objs = len(obj_names)
  for i in range(n_objs - 1):
    obj0 = obj_names[i + 1]
    obj1 = obj_names[i]
    pick_place(obj0, obj1)
```

# Demo


Interactive Demo
Run Example (OpenAI API Key not required)
Set up top-down
Setup/Reset Simulation
Run (this may take 30+ seconds)
Generated Code

OpenAI API Key (this is not stored anywhere): sk-bsmRwMeOcnI [redacted]

Which model? ○ gpt-3.5-turbo-16k ● gpt-4

Which mode? ○ bottom-up ● top-down

Target Task Name (if top-down): build-pyramid-on-pallet

Please see website [https://liruiw.github.io/gensim](https://liruiw.github.io/gensim) and demo [https://huggingface.co/spaces/Gen-Sim/Gen-Sim](https://huggingface.co/spaces/Gen-Sim/Gen-Sim) for details!